Journal of END USER COMPUTING

Vol. 14, No. 3 July-Sept 2002

Table of Contents

User-Developed Applications: Can End Users Assess Quality?

Tanya J. McGill
Murdoch University, Australia

ABSTRACT

Organizations rely heavily on applications developed by end users, yet lack of experience and training may compromise the ability of end users to make objective judgments about the quality of their applications. This study investigated the ability of end users to assess the quality of applications they develop. The results confirm that there are differences between the system quality assessments of end user developers and independent expert assessors. In particular, the results of this study suggest that end users with little experience may erroneously consider the applications they develop to be of high quality. Some implications of these results are discussed.

Keywords: end user computing; user developed applications

INTRODUCTION

User-developed applications (UDAs) form a significant proportion of organizational information systems (McLean, Kappelman, & Thompson, 1993), and the ability to use end user development tools is often a position requirement instead of an individual option (Brancheau & Brown, 1993). The benefits that have been claimed for user development of applications include better access to information and improved quality of information, leading to improved employee productivity and performance. However the realization of these benefits may be put at risk because of problems with information produced by UDAs that may be incorrect in design, inadequately tested, and poorly maintained.

Despite these risks organizations generally undertake little formal evaluation of the quality of applications developed by end users (Panko & Halverson, 1996). In the majority of organizations, the only measures of whether an application is suitable for use are user developers' subjective assessments of their applications. Yet purely subjective, personal evaluations of UDA quality could be at wide variance with actual quality. Lack of experience and training may compromise the ability of end users to make objective judgments about the quality of their applications, but it appears that many end users do lack experience and training in both use of system development tools and in systems development procedures (Cragg & King, 1993).

There has been little empirical research on user development of applications (Shayo, Guthrie, & Igbaria, 1999), and most of what has been undertaken has used user satisfaction as the measure of success because of the lack of objective measures available (Etezadi-Amoli & Farhoomand, 1996). The fact that vital organizational decision making relies on the individual end user's assessment of application effectiveness suggests that more insight is needed into the ability of end users to assess the success of their own applications, and that as well as user satisfaction, additional criteria of success should be considered.

Research on the relationship between experience or training and the success of UDAs has been inconclusive. Some studies have found positive impacts (Crawford, 1986; Nelson & Cheney, 1987; Raymond & Bergeron, 1992) and some have found negative impacts (Amoroso, 1986; Crawford, 1986; Janvrin & Morrison, 2000). Yaverbaum and Nosek (1992) speculated that computer training increases one's expectations of information systems, and hence may actually cause negative perceptions. This may be the case for both training and experience in the UDA domain and may go some way to explaining the lack of conclusive results in the literature.

There have been many calls for the development of more direct and objective measures of UDA effectiveness (Al-Shawaf, 1993; Edberg & Bowman, 1996; Igbaria, 1990; Rivard, Poirier, Raymond, & Bergeron, 1997). There have also been some attempts to move away from the use of user satisfaction as the major indicator of UDA success and to adopt a software engineering approach with a focus on application quality rather than user satisfaction. Edberg and Bowman (1996) compared the quality of UDAs with applications developed by information systems professionals, and found UDAs to be of significantly lower quality. Rivard and her colleagues (Rivard et al., 1997) noted that although the conceptual definitions of quality from the software engineering literature are appropriate for UDAs, the operationalizations in terms of software metrics are not. They therefore attempted to capture both the user perspective and the more technical aspects of UDA quality through a validated assessment instrument to be completed by end user developers (Rivard et al., 1997). However, none of these studies have compared user and expert assessments of UDA quality, nor looked at the roles of experience and training in end users' ability to assess the quality of applications. This paper describes a study which uses direct examination of applications to compare users' and experts' assessments of UDAs.

RESEARCH QUESTIONS

As discussed above, reliance on end user perceptions of UDA quality may be problematic because users may not only

lack the skills to develop quality applications but may also lack the knowledge to make realistic determinations about the quality of applications that they develop. A user developer may be pleased with the quality of their 'creation' and its contribution to their decision-making activities, when in fact the application includes serious errors such as incorrect formulae (Edberg & Bowman, 1996). End user developers who are unaware of quality problems in their applications may make errors in tasks or make poor decisions, which in turn could impact on organizational performance.

The potential for a user developer's perceptions to be colored by ignorance indicates the need for research assessing the ability of end users to evaluate the quality of the products of their own application development work. This can be accomplished by comparing user developers' perceptions of application quality with independent expert assessments.

The primary research question investigated in this study was:

> How do user developer assessments of the quality of applications they have developed differ from independent expert assessments?

As discussed earlier, in previous studies that have related computing experience and training to EUC success, the dependent variable used has mainly been user satisfaction and the results have not been conclusive. While Crawford (1986) found that greater user developer experience was associated with higher levels of satisfaction, Al-Shawaf (1993) did not find any relationship between development experience and user satisfaction, and Amoroso (1986) found that the lower the level of programming skills and report building skills reported, the higher was the satisfaction. Janvrin and Morrison (2000) found that their more experienced subjects were less confident that their applications were error free.

Crawford (1986) also found that higher levels of training were generally associated with lower levels of user satisfaction, while Raymond and Bergeron (1992) found microcomputer training to have a significant positive effect on satisfaction with decision making, and Nelson and Cheney (1987) concluded that there is generally a positive relationship between computer-related training that a user receives and his or her ability to use the computer resource.

Hence in this study the second research question to be answered was:

> How do experience and training influence differences between user developer and independent expert assessments of UDAs?

It was hypothesized that:

1) End user assessments of UDA quality will not be consistent with expert assessments of UDA quality when the user developer has little experience with application development using the chosen tools.
2) End user assessments of UDA quality will not be consistent with expert assessments of UDA quality when the user developer has had little training in use of the chosen tools.

METHOD

The study was conducted with Master's of Business Administration (MBA) students participating in a business policy simulation over a period of 13 weeks as part of a capstone course in Strategic Management. All subjects had at least two years of previous professional employment.

The general applicability of research findings derived from student samples has been an issue of concern. However, Briggs et al. (1996) found MBA students to be good surrogates for executives in studies relating to the use and evaluation of technology, suggesting that the students who participated in this study can be considered as typical of professionals who would be involved in user development of applications in organizations. The opportunity to undertake the study in a relatively controlled environment, where all applications had similar requirements and there was minimum confounding by extraneous variables, was considered worth trading off against the greater generalizability that may have been obtained from a field study.

The Game

The Business Policy Game (BPG) (Cotter & Fritzche, 1995) simulates the operations of a number of manufacturing companies. Participants assume the roles of managers, and make decisions in the areas of marketing, production, financing, and strategic planning. Typical decisions to be made include product pricing, production scheduling, and obtaining finance.

In this study the decisions required for the operation of each company were made by teams with four or five members. Decisions were recorded twice a week and the simulation run immediately afterwards so that results were available for teams to use during the next decision period. Each team was free to determine its management structure, but in general the groups adopted a functional structure, with each member responsible for a different area of decision making. The simulation accounted for 50% of each subject's overall course grade.

The User-Developed Applications

The subjects developed their own decision support systems using spreadsheets to help in their decision making. Decision support systems were developed either by individuals to support their own area of responsibility or by several members of a team. Where several members of a team worked on one application, each was responsible for one worksheet, that relating to their area of responsibility. The unit of the analysis in the study was an individual's application. If they wished, the subjects were able to use simple templates available with the game as a starting point for their applications, but they were not constrained with respect to what they developed, how they developed it, or the hardware and software tools they used. The majority of applications were developed in Microsoft Excel© but some subjects also used Lotus 1-2-3© and Claris Works©. The spreadsheets themselves were not part of the course assessment, so there were no formal requirements beyond students' own needs for the game. The fact that development of applications was optional and unrelated to the purposes of this study reduces the artificiality of the study situation.

Procedure for Data Collection

Each subject was asked to complete a written questionnaire and provide a copy of their spreadsheet on disk after eight 'quarterly' decisions had been made (four weeks after the start of the simulation). This point was chosen to allow sufficient time for the development and testing of the applications. The majority of completed questionnaires and spreadsheets were collected in person during the time when subjects were submitting their decisions, but where this wasn't possible, subjects were sent a

follow-up letter with a reply paid envelope. Ninety-one questionnaires were distributed and 79 useable responses were received giving a response rate of 86.8%.

The Instrument

The questionnaire consisted of two sections. The first section asked questions about the subjects and their previous training and experience with spreadsheets, and the second section asked questions about the spreadsheet they had developed. Spreadsheet experience was measured in years and subjects were subsequently categorized (based on the spread of experience in the sample) as low experience (0–4 years experience), medium experience (5–8 years experience) or high experience (9+ years experience). Previous spreadsheet training was measured using a 4-item, 5-point Likert-type scale from Igbaria (1990) which asked for level of training received in each of four types of training (college or university; vendor; in-company; self study). Scores for the four types of training were summed and subjects were subsequently categorized as low training (score less than 6), medium training (score of 7–9), or high training (score of 10 or more).

System quality relates to the quality of the information system itself and is concerned with matters such as whether or not there are 'bugs' in the system, the consistency of the user interface, and ease of use. In this study system quality was operationalized based upon the instrument developed by Rivard et al. to assess specifically the quality of user-developed applications (Rivard et al., 1997). Rivard et al.'s instrument was designed to be suitable for end user developers to complete, yet to be sufficiently deep to capture their perceptions of components of quality.

Seven of the eight dimensions of quality in Rivard et al.'s instrument could be considered for these applications. These were reliability, effectiveness, portability, economy, user-friendliness, understandability, and maintainability. The verifiability dimension was not included because the processes being examined in the questionnaire items relating to verifiability were not applicable to the environment in which the development was done. A number of individual items were also not included either because they were not appropriate for the applications under consideration (e.g., specific to database applications) or because they were not amenable to expert assessment (e.g., required either privileged information about the subjects' performance in the game or access to the hardware configurations on which the spreadsheets were originally used). Minor adaptations to wording were also made to reflect the terminology used in the BPG and the environment in which application development and use occurred.

The resulting system quality scale consisted of 40 items, each scored on a Likert scale of 1 to 7 where (1) was labeled 'strongly agree' and (7) was labeled 'strongly disagree' (see Appendix 1 for a list of the items). Measures for each of the quality dimensions were obtained by averaging the values of the criterion variables relating to that dimension. An overall application quality measure was obtained by averaging the seven quality dimension scores. This is consistent with the approach used by Rivard et al. The instrument had a Cronbach's alpha of 0.82.

Independent Expert Assessment of System Quality

Two independent assessors using the same set of items also assessed the sys-

tem quality of each UDA. Both assessors were information systems academics with years of experience teaching spreadsheet design and development. Before assessing the study sample, the assessors spent a substantial amount of time familiarising themselves with the BPG and then completed four pilot evaluations of applications not included in the study sample. Differences were discussed and adjustments made to ensure consistency between the assessors. Assessments of the actual UDAs were then undertaken. The quality ratings of the two independent assessors were highly correlated ($r = 0.73$, $p = 0.000$).

RESULTS

Of the 79 subjects 78.5% were male and 21.5% female (62 males, 17 females). Their ages ranged from 21 to 49 with an average age of 31.8. Subjects reported an average of 5.9 years experience using spreadsheets (with a range from 0 to 15 years).

Table 1 indicates that the subjects had received relatively little spreadsheet training. More than 50% of the subjects had received no in-company or vendor training and just under 50% had received no college or university training. Self-study was the predominant means by which students had acquired their knowledge of spreadsheets.

The first research question considered how end user developer assessments of application quality might differ from those of the independent experts. To address this question, the mean scores for each quality dimension as assessed by the user developers were compared with the independent assessments (Table 2). The scores for each quality dimension as assessed by the user developers were compared statistically with the independent assessments using paired sample t-tests.

There were significant differences on five of the quality dimensions. The user developers rated the effectiveness and portability of their applications significantly lower than did the independent assessors ($t=-2.67$, $p=0.009$; $t=-3.55$, $p=0.001$) and rated reliability, understandability, and user-friendliness significantly higher than did the independent assessors ($t=7.25$, $p=0.000$; $t=4.58$, $p=0.000$; $t=4.06$, $p=0.000$). However, the overall assessments of quality were not found to be significantly different as the above differences canceled out. The rankings of mean quality across the dimensions were also considered. The applications were ranked highest on portability and lowest on reliability by both the user developers and the independent assessors, but the other dimensions were ranked differently. A Spearman rank order correlation test showed the rankings to be not significantly correlated ($rho = 0.607$, $p = 0.148$).

Table 1: Summary of the Subjects' Previous Spreadsheet Training

Training Source	Mean	Level of Training									
		Number in each category									
		(1) None		(2)		(3)N		(4)		(5) Extr. Intensive	
		N	%	N	%	N	%	N	%	N	%
College or University	2.0	46	58.2	8	10.1	6	7.6	11	13.9	7	8.9
Vendor	1.5	62	78.5	3	3.8	4	5.1	5	6.3	4	5.1
In-company	1.7	52	65.8	6	7.6	12	15.2	7	8.9	1	1.3
Self study	3.3	8	10.1	8	10.1	26	32.9	23	29.1	13	16.5

Table 2: A Comparison of the Mean User Developer Assessments of Each Quality Dimension with the Independent Expert Assessments for Each Quality Dimension

Quality dimension	User developer assessment			Independent expert assessment			Significance
	Mean	SD	Ranking	Mean	SD	Ranking	
Economy	3.85	1.75	2	4.27	0.71	3	p=0.058
Effectiveness	3.77	1.29	5	4.29	1.03	2	p=0.009
Maintainability	3.56	1.44	6	3.29	1.25	4	p=0.228
Portability	3.91	1.31	1	4.51	0.68	1	p=0.001
Reliability	3.06	0.90	7	2.19	0.65	7	p=0.000
Understandability	3.83	0.83	3	3.20	0.71	5	p=0.000
User-friendliness	3.81	0.94	4	3.18	0.81	6	p=0.000
Overall quality	3.68	0.80		3.57	0.60		p=0.380

Several individual questionnaire items stood out in illustrating problems that many end user developers had in recognizing quality problems with their applications. These are shown in Table 3 below. If end user developers have serious misconceptions such as these, it could pose significant risks to the security and integrity of organizational data and to the quality of organizational decision making.

The second research question considered whether experience and training might influence differences between user developer and independent expert assessments of user-developed applications. The role of experience was considered first. End user developers were categorized according to the number of years of spreadsheet experience they had: low experience (0–4 years; N = 29), medium experience (5–8 years; N = 29), and high experience (9+ years; N = 21). Table 4 shows the mean end user quality assessments of the applications for the three experience groupings, the mean independent assessments, and also the mean difference between the end user developer and independent assessment for each application.

In order to analyze the differences in quality assessments between end users with different experience levels these were compared across the groups using

Table 3: System Quality Instrument Items on Which There Were Major Differences of Opinion

	% of applications for which end user developers agreed	% of applications for which expert assessors agreed
Unauthorized users could not easily access all the data or a part of it	35.4	16.7
Each user owns a unique password	29.5	9.0
This system automatically corrects certain types of errors at data-entry time	35.0	0.0
This system always issues an error message when it detects an error	26.0	0.0
The system performs an automatic backup of the data	26.3	0.0
The system never modifies a cell without asking for a confirmation and getting a positive response	32.9	5.1

Table 4: A Comparison of the Assessments of Each Quality Dimension Across the Low, Medium and High Experience Groups

Quality dimension	*Low Experience*		*Med. Experience*		*High Experience*		*Significance*
	Mean	*Std. dev.*	*Mean*	*Std. dev.*	*Mean*	*Std. dev.*	
Economy							
End user developer	4.03	1.64	3.86	1.50	3.57	2.18	0.654
Expert assessors	4.24	0.73	4.16	0.77	4.48	0.58	0.294
Difference	-0.21	1.94	-0.30	1.80	-0.90	2.25	0.433
Effectiveness							
End user developer	4.07	1.19	3.41	1.32	3.85	1.27	0.141
Expert assessors	4.24	1.04	4.07	1.21	4.69	0.58	0.103
Difference	-0.17	1.64	-0.68	2.01	-0.82	1.40	0.367
Maintainability							
End user developer	3.75	1.55	3.86	1.09	2.88	1.52	0.037 [LH, MH]
Expert assessors	3.14	1.24	3.26	1.35	3.58	1.10	0.450
Difference	0.63	2.00	0.62	1.86	-0.70	1.53	0.022 [LH, MH]
Portability							
End user developer	4.02	1.43	3.79	1.09	3.83	1.49	0.797
Expert assessors	4.41	0.89	4.54	0.49	4.59	0.56	0.650
Difference	-0.40	1.66	-0.68	1.19	-0.76	1.62	0.652
Reliability							
End user developer	3.31	0.82	3.13	0.94	2.66	0.87	0.040 [LH]
Expert assessors	2.20	0.69	2.06	0.64	2.34	0.61	0.329
Difference	1.11	0.96	1.02	1.14	0.32	0.90	0.018 [LH, MH]
Understandability							
End user developer	4.16	0.69	3.80	0.74	3.45	0.98	0.011 [LH]
Expert assessors	3.18	0.68	3.12	0.83	3.37	0.61	0.476
Difference	1.02	1.06	0.66	1.23	0.08	1.23	0.026 [LH]
User-friendliness							
End user developer	3.95	1.05	3.92	0.71	3.47	1.00	0.145
Expert assessors	3.12	0.83	3.18	0.90	3.28	0.66	0.808
Difference	0.83	1.54	0.73	1.14	0.19	1.29	0.225
Overall quality							
End user developer	3.89	0.79	3.68	0.68	3.38	0.90	0.086
Expert assessors	3.50	0.62	3.48	0.64	3.76	0.49	0.221
Difference	0.38	1.06	0.20	1.11	-0.38	0.99	0.043 [LH]

[LH]Significant difference in means ($p<0.05$) between the low experience and high experience groups
[MH]Significant difference in means ($p<0.05$) between the medium experience and high experience groups

ANOVA. In cases where ANOVA indicated significance differences the Bonferroni test was used to perform pairwise comparisons to determine the exact nature of the difference. The results provide support for Hypothesis 1.

The end user developers with low experience considered their applications to be of higher quality on all dimensions than did the end user developers with high experience. There were significant differences in end user quality assessments across the experience groups for maintainability ($F=3.45$, $p=0.037$), reliability ($F=3.36$, $p=0.040$), and understandability ($F=4.80$, $p=0.011$). In each of these cases Bonferroni tests showed that the quality assessments for the applications of the low experienced end users were significantly higher than those of the high experienced end user developers. However, no differences between the experience groupings were found in the independent assessments of quality.

A comparison of the difference scores between the three groups further

supports what the results of the end user assessment comparison suggested. On each dimension and on overall quality the low experience group had either the largest positive difference or least negative difference. This pattern of differences suggests that the low experience group perceived applications of equivalent quality (as assessed independently) to be of better quality than did the high experience group. That is, they tended to overestimate the quality of their applications relative to the high experience group. The differences between groups were significant for overall quality (F=3.27, p=0.043) and for the following quality dimensions: maintainability (F=4.02, p=0.022), reliability (F=4.24, p=0.018), and understandability (F=3.84, p=0.026).

The role of training in the ability of an end user developer to make objective assessments of application quality was considered by comparing the three groupings of end user developers: low training (training score <=6; N=20), medium training (score 7 – 9; N=35), and high training (score >=10; N=23). Table 5 shows the mean quality assessments by the end user developers, the independent assessors, and also the mean difference between the end user developer and independent assessment for each application. In order to analyze the differences in quality assessments between end users with different training levels, these

Table 5: A Comparison of the Assessments of Each Quality Dimension Across the Low, Medium and High Training Level Groups

Quality dimension	*Low Training*		*Med. Training*		*High Training*		*Significance*
	Mean	*Std. dev*	*Mean*	*Std. dev*	*Mean*	*Std. dev*	
Economy							
End user developer	3.90	1.65	3.77	1.82	3.83	1.75	0.966
Expert assessors	4.32	0.67	4.20	0.75	4.32	0.70	0.761
Difference	-0.42	1.84	-0.43	2.16	-0.50	1.89	0.990
Effectiveness							
End user developer	3.75	1.33	3.62	1.28	3.96	1.26	0.624
Expert assessors	4.15	1.05	4.30	1.09	4.41	0.95	0.722
Difference	-0.40	1.81	-0.66	1.79	-0.45	1.66	0.844
Maintainability							
End user developer	3.74	1.22	3.27	1.56	3.78	1.38	0.330
Expert assessors	3.40	1.15	3.01	1.36	3.65	1.07	0.152
Difference	0.35	1.76	0.26	2.13	0.15	1.72	0.942
Portability							
End user developer	3.75	1.36	3.99	1.27	3.78	1.37	0.768
Expert assessors	4.31	1.04	4.58	0.55	4.57	0.41	0.338
Difference	-0.56	1.49	-0.59	1.56	-0.70	1.41	0.946
Reliability							
End user developer	3.16	0.93	2.82	0.70	3.33	1.08	0.097
Expert assessors	2.20	0.63	2.15	0.69	2.22	0.64	0.919
Difference	0.95	1.04	0.67	0.87	1.07	1.32	0.355
Understandability							
End user developer	4.07	0.69	3.78	0.75	3.70	1.03	0.337
Expert assessors	3.18	0.66	3.19	0.75	3.26	0.76	0.918
Difference	0.94	1.02	0.59	1.03	0.40	1.60	0.363
User-friendliness							
End user developer	3.93	0.80	3.74	0.93	3.79	1.10	0.762
Expert assessors	3.26	0.77	3.16	0.95	3.16	0.62	0.902
Difference	0.67	1.10	0.57	1.55	0.61	1.30	0.967
Overall quality							
End user developer	3.74	0.75	3.57	0.76	3.74	0.88	0.642
Expert assessors	3.55	0.63	3.51	0.63	3.66	0.56	0.688
Difference	0.19	1.05	0.05	1.13	0.08	1.13	0.902

were compared across the groups using ANOVA. The results do not provide support for Hypothesis 2 as no significant differences were found between end users with low, medium, and high levels of training with respect to end user developer quality ratings, independent quality ratings, or difference scores on any of the quality dimensions. However it is interesting to note that the difference scores showed a similar (though not significant) pattern to the difference scores for the experience groupings, with the low experience group having larger positive or less negative scores on all dimensions but reliability.

DISCUSSION

This study investigated the ability of end users to assess the quality of the applications they develop. The results indicate that there are some differences between the system quality assessments of end user developers and independent expert assessors, and also differences between quality assessments of end users with low and high levels of experience. In particular, the results of this study suggest that end-user developers with little experience may rate applications of equivalent quality more highly than do experienced user developers. While the findings should be considered preliminary because of the use of MBA student subjects, the results raise concerns about the heavy reliance of organizations on users' perceptions of their own applications. These findings should be followed up by field studies evaluating UDAs in organizations.

Can User Developers Assess the Quality of Their Applications?

User developer assessments of overall application quality were not found to be significantly different from the independent assessments. This is because some of the differences at the quality dimension level are in different directions and partially cancel out. There were significant differences on five of the quality dimensions. The user developers rated the effectiveness and portability of their applications significantly lower than did the independent assessors. It is interesting that the user developers were more critical with respect to the effectiveness of applications than the independent assessors were. Of all the quality dimensions considered, effectiveness was the dimension of most immediate importance to the user developers, and the dimension about which they received the most feedback via the BPG reports, and hence it is the dimension about which they could be expected to be most critical.

The questionnaire items on portability related to two criteria: portability across hardware, and portability across organizational environments. User developer assessments differed significantly from the independent assessments only with respect to portability across different hardware platforms. This appears to result from a lack of awareness of just how portable applications developed in Microsoft Excel© currently are. The fact that both the end user developers and the independent assessors ranked portability highest among the dimensions suggests that the difference is not too problematic.

The user developers rated the reliability, understandability, and user-friendliness of their applications significantly higher than did the independent assessors. Spreadsheets are the first introduction to application development for many end users, and in general end users have not been trained in systems analysis and design, and tend to overlook issues such as reliability and auditability (Ronen, Palley, & Lucas, 1989). The differences in reliability and understandability assessments are consistent with

the findings of Nelson (1991), who identified the major skill deficiencies of end users as being in technical and information system product areas, and with those of Edberg and Bowman (1996) who found major data integrity problems with the end user applications in their study. Rivard et al. (1997) noted that they would not be surprised to find user attitudes quite impervious to the more technical dimensions of application quality, as the more 'technical' dimensions of quality would be expected to preoccupy computer professionals, but probably not end users unless they have been trained to focus on them. However the fact that reliability was the lowest ranking dimension for user developers as well as the independent assessors provides some hope that user developers are gaining insight into the weaknesses of their applications.

The difference in assessments of user-friendliness between the user developers and the independent assessors could be because the familiarity user developers gain with their applications during development may color their perceptions of their application's user-friendliness. As many UDAs are used by end users other than the developer (Bergeron & Berube, 1988), this could cause problems.

There might also be a self-evaluation effect. Doll and Torkzadeh (1989) speculated on the 'bias' of end user developers but didn't empirically investigate it. McGill et al. (1998) provided preliminary evidence of this 'bias' in the UDA domain by showing that end users exhibited an increased degree of satisfaction with a spreadsheet application that they had developed themselves compared with another end user using the same application.

The Effect of Experience

Level of spreadsheet experience appeared to play an important role in the ability of end user developers to assess system quality. Those end users with little experience rated the quality of their applications higher on all dimensions than did the user developers in the high experience group. The differences between end user assessments of quality and independent assessments were also either larger (if positive) or less negative, for the low experience group. This suggests that lack of experience seriously impedes the ability of user developers to be objective about the quality of their applications. The quality dimensions for which the differences between experience levels were significant were the more technical dimensions of maintainability, reliability, and understandability. It seems that despite Rivard et al.'s (1997) concerns about end user awareness of the technical dimensions of quality, with experience comes some increase in awareness.

It is interesting to note that no relationship was found between level of spreadsheet experience and the independent expert quality assessments. Those with more experience did not develop higher quality applications. Perhaps despite being more aware of the limitations of their applications, they did not aim to develop quality applications. This could suggest a lack of awareness of the consequences of using applications of low quality (Ronen et al., 1989). A lack of concern for consequences might be exacerbated by two factors in this study. Firstly, the applications did not form part of the formal assessment for the course, and secondly, the subjects were aware that the applications would only be required for a limited period of time (the duration of the simulation). However these circumstances are often mirrored in the workplace with no external controls being placed on developments and with end users developing applications that they believe will only be used once and then using

them repeatedly (Kroenke, 1992). It can only be hoped that despite the fact that their applications were not of significantly better quality, the additional insight into the quality of their applications would lead high experience end users to treat their results with more caution.

The Effect of Training

In this study, level of spreadsheet training did not appear to play a role in determining either the ability of end user developers to assess system quality, or system quality itself. The differences between the end user developer perceptions of quality and the independent assessments were not significantly lower for those end user developers in the highest training group. Both the amount of training that the subjects had received and the types of training could explain the results. As Table 1 shows, the subjects had received relatively little training and the major means of training was self-study. It has been suggested that when end users are self-taught, the emphasis is predominantly on how to use the software rather than broader analysis and design considerations (Benham, Delaney, & Luzi, 1993). Thus the subjects in this study may not have received training of a type conducive to reflection on system quality. As self-training has been shown to be the major form of training in a number of studies (e.g., Amoroso & Cheney, 1991; Benham et al., 1993; Chan & Storey, 1996), the results of this study may highlight potential problems in a wide range of organizations.

The fact that no relationship was found between amount of previous spreadsheet training and the independent quality assessments may also relate to the amounts and types of training received. Preliminary results of Babbitt, Galletta, and Lopes's (1998) study of spreadsheet development by novice users suggested that end users whose training emphasizes planning and testing of spreadsheets will develop better quality spreadsheets. However it is also possible that despite the training the subjects may previously have had, they did not consider it important to develop applications of high quality. Future research should investigate the role of type of training in both application quality and end user perceptions of application quality.

CONCLUSION

The results of this study cast some doubts on the ability of end users to make realistic determinations of the quality of applications they develop. Those subjects with little experience erroneously considered their applications to be of higher quality than subjects with more experience did. This may compromise the effectiveness of end users as application developers and could have major consequences when the systems developed are used to support decision making in organizations. Also of concern is the fact that no relationship was found between spreadsheet experience or training and the independent assessments of quality. Those user developers who would be expected to be more realistic in assessing the quality of their applications were, however, not developing applications of higher quality.

Given the increasing importance of user-developed applications to organizational decision making, it is essential that organizations be aware of the potential problems and that steps are taken to address them. Organizations must recognize that end user developers may perceive the information from an application to be suitable to support decision making when, in fact, technical design and implementation flaws have introduced serious errors.

With the majority of organizations imposing no quality control procedures on user developers (Bergeron & Berube, 1990; Cale, 1994; Panko & Halverson, 1996), a number of authors have suggested that training may be the most effective tool for minimizing risks associated with end user computing (Cragg & King, 1993; Edberg & Bowman, 1996; Nelson, 1991). However, as the results of this study show, increasing levels of training is no guarantee of improvements in quality. Future research should target the role of training that emphasizes application development methods and procedures, especially in the area of quality assurance. Intelligent application development tools such as the one proposed by Shah and Lawrence (1996) could also increase the quality of UDAs by embedding the necessary knowledge about the more technical aspects of system quality. Unless proficiency in developing applications is increased, organizations risk incurring considerable costs.

APPENDIX 1

Questionnaire Items Used to Measure System Quality

Economy
- The system increased my data processing capacity

Effectiveness
- The system provides all the information it should

Maintainability
- This system provides the capability to import data from other applications
- It is possible to copy parts of the system (outputs or data) into other systems or to link with other systems

Portability
- The system can be run on computers other than the one presently used
- The system could be used in other similar organisational environments, without any major modification

Reliability
- Unauthorised users could easily access all the data or a part of it
- Each user owns a unique password
- Unauthorised access is controlled in several parts of the system
- Errors in the system are easy to identify
- Each password limits the access to specific parts of the system
- This system (rather than the spreadsheet package) automatically corrects certain types of errors, at data-entry time
- Should an error arise, the system provides the capability to perform some checking in order to locate the source of error
- This system (rather than the spreadsheet package) always issues an error message when it detects an error
- All outputs provided by this system are required
- The data entry sections provide the capability to easily make corrections to data
- Outputs provided by this system are comprehensive
- The system contains all the information required to produce comprehensive outputs
- The system does not destroy any information without asking for a confirmation and getting a positive response
- The system provides default values in the data-entry section
- The system performs an automatic backup of the data
- Data is labelled so that it can be easily matched with other parts of the system

- The system never modifies a cell without asking for a confirmation and getting a positive response
- Corrections to errors in the system are easy to make

Understandability

- The same terminology is used throughout the system
- Data entry sections are organised in such a way that the data elements are logically grouped together
- The data entry areas clearly show the spaces reserved to record the data
- The format of a given piece of information is always the same, wherever it is used in the system
- Headings provide information related to the nature of data in the system (e.g., empno = employee number)
- The system is broken up into separate and independent sections
- Each section has a unique function
- Each section includes enough information to help you understand its functioning
- The documentation provides all the information required to use the system
- Message presentation is always the same (position, terminology, style)
- The documentation explains the functioning of the system

User-Friendliness

- Using the system is easy, even after a long period of non-utilisation
- The system is easy to learn by new users
- The outputs are easy to understand
- The terms used in data-entry sections are familiar to users
- Queries are easy to make

REFERENCES

Al-Shawaf, A.-R. H. (1993). *An Investigation of the Design Process for End-User Developed Systems: An Exploratory Field Study.* Unpublished PhD, Virginia Commonwealth University.

Amoroso, D. L. (1986). *Effectiveness of End-User Developed Applications in Organizations: An Empirical Investigation.* Unpublished PhD, University of Georgia.

Amoroso, D. L. & Cheney, P. H. (1991). Testing a causal model of end-user application effectiveness. *Journal of Management Information Systems, 8*(1), 63-89.

Babbitt, T. G., Galletta, D. F. & Lopes, A. B. (1998). Influencing the success of spreadsheet development by noviceuUsers. *Proceedings of the Nineteenth International Conference on Information Systems*, 319-324.

Benham, H., Delaney, M. & Luzi, A. (1993). Structured techniques for successful end user spreadsheets. *Journal of End User Computing*, (Spring), 18-25.

Bergeron, F. & Berube, C. (1988). The management of the end-user environment: An empirical investigation. *Information & Management, 14*, 107-113.

Bergeron, F. & Berube, C. (1990). End users talk computer policy. *Journal of Systems Management,* (December), *14-16*, 32.

Brancheau, J. C. & Brown, C. V. (1993). The management of end-user computing: Status and directions. *ACM Computing Surveys, 25*(4), 450-482.

Briggs, R. O., Balthazard, P. A. & Dennis, A. R. (1996). Graduate business students as surrogates for executives in the evaluation of technology. *Journal of End User Computing, 8*(4), 11-17.

Cale, E. G. (1994). Quality issues for end-user developed software. *Journal of Systems Management,* (January), 36-39.

Chan, Y. E. & Storey, V. C. (1996). The use of spreadsheets in organizations: Determinants and consequences. *Information & Management, 31*, 119-134.

Cotter, R. V. & Fritzche, D. J. (1995). *The Business Policy Game*. Englewood Cliffs, NJ: Prentice-Hall.

Cragg, P. G. & King, M. (1993). Spreadsheet modelling abuse: An opportunity for OR? *Journal of the Operational Research Society, 44*(8), 743-752.

Crawford, J. B. (1986). *An Investigation of Strategies for Supporting and Controlling User Development of Computer Applications*. Unpublished PhD, University of California, Irvine.

Doll, W. J. & Torkzadeh, G. (1989). A discrepancy model of end-user computing involvement. *Management Science, 35*(10), 1151-1171.

Edberg, D. T. & Bowman, B. J. (1996). User-developed applications: An empirical study of application quality and developer productivity. *Journal of Management Information Systems, 13*(1), 167-185.

Etezadi-Amoli, J. & Farhoomand, A. F. (1996). A structural model of end user computing satisfaction and user performance. *Information & Management, 30*, 65-73.

Igbaria, M. (1990). End-user computing effectiveness: A structural equation model. *OMEGA, 18*(6), 637-652.

Janvrin, D. & Morrison, J. (2000). Using a structured design approach to reduce risks in end user spreadsheet development. *Information & Management, 37*(1), 1-12.

Kroenke, D. (1992). *Management Information Systems*. Watsonville, CA: McGraw-Hill.

McGill, T. J., Hobbs, V. J., Chan, R. & Khoo, D. (1998). User satisfaction as a measure of success in end user application development: An empirical investigation. In Khosrowpour, M. (Ed.), *Proceedings of the 1998 IRMA Conference*, pp. 352-357. Hershey, PA: Idea Group Publishing.

McLean, E. R., Kappelman, L. A. & Thompson, J. P. (1993). Converging end-user and corporate computing. *Communications of the ACM, 36*(12), 79-92.

Nelson, R. R. (1991). Educational needs as perceived by IS and end-user personnel: A survey of knowledge and skill requirements. *MIS Quarterly, 15*(4), 503-525.

Nelson, R. R. & Cheney, P. H. (1987). Training end users: An exploratory study. *MIS Quarterly, 11*, 547-559.

Panko, R. R. & Halverson, R. P. (1996). Spreadsheets on trial: A survey of research on spreadsheet risks. *Proceedings of the Twenty-Ninth Hawaii International Conference on System Sciences, 2*, 326-335.

Raymond, L. & Bergeron, F. (1992). Personal DSS success in small enterprises. *Information & Management, 22*, 301-308.

Rivard, S., Poirier, G., Raymond, L. & Bergeron, F. (1997). Development of a measure to assess the quality of user-developed applications. *The DATA BASE for Advances in Information Systems, 28*(3), 44-58.

Ronen, B., Palley, M. A. & Lucas, H. C. (1989). Spreadsheet analysis and design. *Communications of the ACM, 32*(1), 84-93.

Shah, H. U. & Lawrence, D. R. (1996). A study of end user computing and the provision of tool support to advance end user empowerment. *Journal of End User Computing, 8*(1), 13-21.

Shayo, C., Guthrie, R. & Igbaria, M. (1999). Exploring the measurement of end user computing success. *Journal of End User Computing, 11*(1), 5-14.

Yaverbaum, G. J. & Nosek, J. (1992). Effects of information system education and training on user satisfaction. *Information & Management, 22*, 217-225.

Tanya McGill *is a Senior Lecturer in the School of Information Technology at Murdoch University in Western Australia. She has an MBA from the University of Western Australia and is currently completing a PhD at Murdoch University. Her research interests include end user computing and information technology education. Her work has appeared in various journals, including the* Journal of Research on Computing in Education, European Journal of Psychology of Education, *and* Journal of the American Society for Information Science.